

The Unmanned Aerial Vehicle Routing Problem with Recharging

Huiting Mao, Jianmai Shi, Zhongbao Zhou and Long Zheng

Abstract—The application of Unmanned aerial vehicles (UAVs) in both civilian and military domains is drawing increasing attention recently. This paper investigates a new routing problem of small UAVs for information collection, where UAVs can be recharged at platforms (ground vehicles or stations) distributed in the area. Different from the previous works on UAV routing, the UAVs are allowed to partially recharge their batteries according to the requirement in the following route. A mixed integer nonlinear programming model is developed to formulate the problem, where both the overall time for completing all targets' observation and the number of UAVs are minimized. An improved adaptive large neighborhood search (ALNS) algorithm with simulated annealing criterion is designed, and a recharging platform insertion heuristic is developed to determine the recharging strategy and construct feasible solutions. To verify the effectiveness of the proposed ALNS algorithm, a set of new benchmark instances are designed based on the well-known Solomon dataset and solved. The computational results are compared with those obtained by the ant colony optimization and variable neighborhood search, which shows that ALNS performs significantly better and stable. Furthermore, analysis of the experimental results indicates that many advantages can be obtained through introducing the recharging strategy for small UAVs.

Index Terms—Unmanned aerial vehicle, routing, heuristic, recharging

I. INTRODUCTION

SMALL Unmanned aerial vehicles (UAVs) or drones play an increasing role in both civilian and military areas, for instance, agriculture monitoring, disaster relief, battlefield reconnaissance, border patrol and logistic delivery. In the civilian applications, UAVs showed great market potential which leads to significant cost savings in the last-mile package delivery, information collection, wild search & rescue, and agriculture monitoring etc [1]. Many companies around the world including UPS [2], DHL [3], Alibaba [4] and Amazon [5] have adopted UAVs in the “last-mile-delivery” so as to reduce the logistics cost and increase the distribution efficiency. Since UAVs can access targets in dangerous environment without losses of human lives, they are widely used in military

operations. Small UAVs can fly at low altitude and hover over the targets to collect accurate information, and also the advantages in miniaturization, strong concealment, easy maintenance and low cost facilitate their application on Intelligence, Surveillance and Reconnaissance (ISR) missions [6].

Due to the limited capacity of battery power, the small UAVs' endurance range is relatively small, which is viewed as a main barrier of their application on more ISR missions in large areas. Especially when the small UAV has to travel in a large area, the UAV need to recharge its battery during the flying route. Many researches focused on extending the endurance range of small UAVs have been conducted, e.g. new energy supported design and the solar power system [7], automated battery swapping and recharging [8], and efficient power allocation [9].

In order to cope with the limitation of UAV's endurance, we proposed a new application mode of small UAVs, where their battery can be recharged at some platforms distributed in the area. Recently, dynamic wireless charging (DWC) technology [10] is applied as a novel way of recharging electric vehicles, and the battery of EVs can be recharged remotely while it is staying around the infrastructure equipped with DWC. With the miniaturization and intelligence of wireless charging equipment, fast wireless charging devices can be installed on many stations and ground vehicles, which make them act as recharging platforms for small UAVs. When the battery power of small UAV is insufficient, it can visit the near recharging platform to recharge the battery and then continue to carry out the mission. In this case, the routing of UAVs should consider the decisions on the selection of recharging platform and the recharging level of the battery.

The UAV routing problem with recharging has some similarity to the electric vehicle routing problem (EVRP) in commercial field [11]. An important characteristic of EVRP is that EVs need to visit the recharging station during the route to extend the endurance range and complete all the delivery tasks. The main difference is that the EVs do not consume the battery power when waiting and serving customers, while small UAVs consume battery power when waiting and collecting information above the target, and the power consumption speed

The research is supported by the National Natural Science Foundation of China (grant no. 71771215) and the Natural Science Fund for Distinguished Young Scholars of Hunan Province (2018JJ1035).

H. Mao and J. Shi are with the Science and Technology on Information Systems Engineering Laboratory, College of System Engineering, National University of Defense Technology, Changsha, 410073, P. R. China. (e-mail: mht126carrie@126.com; jianmaishi@gmail.com).

* Jianmai Shi is the corresponding author.

Z. Zhou is with School of Business Administration, Hunan University, Changsha, P. R. China. (e-mail: Z.B.Zhou@163.com).

L. Zheng is with Technical Service Center of Military Vocational Education, National University of Defense Technology, Changsha, 410073, P. R. China. (e-mail: zhenglong@nudt.edu.cn).

is usually faster when UAV is collecting information due to additional power consumption by carried sensors.

In this paper, we investigated the UAV routing problem with recharging and established a mixed integer nonlinear programming (MINLP) model to formulate the problem. An improved adaptive large neighborhood search (ALNS) embedded with a recharging platform insertion strategy is proposed to find the global better solutions for the problem. Moreover, we designed a set of benchmark instances based on the dataset in [12], and the performance of the proposed ALNS algorithm is tested. Also, the performance and efficiency of the ALNS algorithm is compared with the ant colony optimization (ACO) and variable neighborhood search (VNS) algorithms. Computational results show that the ALNS can achieve much better solutions in shorter computation time for most of the instances.

The reminder of this paper is organized as follows: In the second section, related literatures are reviewed. In the third section, the problem is illustrated and formulated as a MINLP model. To find the feasible solution in a short time for large instances, an improved ALNS algorithm is proposed in Section IV. The performance of the ALNS algorithm is tested by newly designed benchmark dataset in the 5th section. Finally, the whole work is summarized and future directions are discussed.

II. LITERATURE REVIEW

In this section, the two streams of relevant literatures are reviewed, which are the UAV routing problem and the EVRP.

UAV routing problems have been investigated in the Military and Civilian application domains. Shetty et al. [13] proposed a tactical routing problem of a team of UAVs to conduct attacking mission based on the priorities of targets, where the UAV can carry ammunition to attack different targets. Ceccarelli et al. [14] investigated a micro UAV routing problem for reconnaissance and carried on the simulation experiments for finding the robust solution in the presence of randomly perturbed wind. Mufalli et al. [15] considered the simultaneous sensors selection and routing of UAVs where the loads are the influencing factors of endurance range and a new mathematical model is constructed to solve the problem. Avellar et al. [16] routed a group of UAVs for area coverage to optimize the minimum coverage time, where intelligence collection can be carried out in the fixed area and the specified time windows of targets are taken into consideration. Evers et al. [17] studied the routing problem of multiple UAVs for reconnaissance mission, taking the uncertainty in the fuel usage into account. Mahmud and Cho [18] investigated the UAV routing problem where UAVs need to avoid being predicted by the enemy, and Vanegas et al. [19] considered the routing problem of UAVs in the environmental uncertainty. In civilian application, UAVs can be used for logistics delivery [20], medical material transportation [21], efficient road detection and tracking [22] and disaster relief operations [23]. Kevin et al. [24] investigated the vehicle routing problem for drone delivery scenarios which minimizes both the cost and the overall delivery time considering an energy consumption model. Sawadsitang et al. [25] proposed the joint ground and aerial delivery service

optimization and planning framework considering the uncertainty of drone package delivery. To solve the problem, they formulated a three-stage stochastic integer programming model with a decomposition method. In the work [26], the UAVs were used for dynamic wildfire tracking since the UAVs can work in hazardous fire tracking instead of humans, and a distributed control framework was proposed for the UAV team.

However, due to the endurance range of small UAV, the radius of its endurance range is restricted, which greatly limited their applications in large areas. To overcome this difficulty, Liu et al. [6] and Luo et al. [20] proposed a novel two-echelon ground vehicle (GV) and UAV cooperated routing problem (2E-GUCRP) for ISR missions, where the GV serves as the mobile platform carrying the UAVs and recharging the UAV's battery. In this new mode, UAV can enlarge its endurance range through multiple starting on the GVs. In recently years, the application of UAVs in the civilian operation is drawing increasing attention, many commercial logistic companies used UAVs for the 'last-mile-delivery' to save the cost for logistic distribution. Chiang et al. [27] proposed a GV and UAV cooperated system where GVs are the mobile platform for UAVs to start and land on, and both of them can deliver the packages during the route. To overcome the limited endurance range of the small UAVs, Sungwoo and Ilkyeong [28] investigated a truck-drone system, where a fixed drone station is built to collect drones and recharging equipments, and a truck is employed to connect the station and the logistics distribution center. Liu et al. [29] designed a Simulated Annealing algorithm (SA) to solve the 2E-GUCRP problem for package delivery. In these studies, with the assistance of ground vehicles, the endurance range of UAV is effectively extended. However, UAV can only visit targets which are restricted by the path of GV. To extend the endurance range and release the UAV's dependence on GV, Coelho et al. [30] designed a two-level routing problem model, where the UAV can be recharged at a given recharging station to complete the tour. Li et al. [31] proposed a mission planning method which considered both the routing of UAVs and recharging stations location, and assumed the recharging procedure takes a fixed time. Ribeiro et al. [32] studied the application of UAVs for belt conveyor inspection system in the mining industry where UAVs are restricted to be fully charged at the recharging station. To extend the UAV mission coverage, Nouredine et al. [33] utilized the public land transport vehicle carrying the UAV during part of its route for saving the energy consumption, and the recharging power is set to a fixed parameter. Yu et al. [34] studied the UAV route planning by allowing it to visit the fixed recharging stations and the unmanned ground vehicles (UGVs) serve as mobile stations.

The routing problem with recharging is also studied in the EVRP. Schneider et al. [12] studied the EVRP with Time Windows (EVRPTW), where the full fast recharging strategy is applied. They designed a metaheuristic through integrating the VNS algorithm with Tabu search, which is tested by instances generated from the Solomon dataset. Ding et al. [35] studied the EVRP where the partial recharging for electric vehicles (EVs) is allowed and the capacity of the recharging station is taken into consideration. Keskin and Çatay [36] investigated the

EVRPTW in which EVs are also allowed partially recharged. More works on EVRP can be found in the comprehensive review by [37].

From the related literature presented above, we can see that it is an important research topic to investigate different recharging strategies for enlarging the endurance of UAV. The mode of UAVs recharged by smart wireless recharging platforms is a new research area, and more efficient models and algorithms are required.

III. PROBLEM FORMULATION

The initial motivation of the UAV routing problem with recharging is from an application of reconnaissance missions in battlefield, where multiple small UAVs start from the base and collect information at a set of targets. There are a certain number of ground (combat) vehicles distributed in the battlefield, which are configured with fast wireless charging devices and act as recharging platforms of UAVs. Before the UAV's battery powers off, it can fly to any of these platforms for recharging and then continue to visit the following targets. Fig. 1 presents an illustrative example of the problem. The different battery icons denote the power level of the UAV's battery after visiting each target. The UAV start from the unique base with a fully charged battery, and after visiting target T3, the battery power cannot support it to visit the next target T4, and therefore the UAV finds the nearby recharging platform P2 for recharging its battery partially. After recharging, the UAV continues its tour for visiting T4 and T5, and then flies back to the base. Furthermore, the UAV is allowed to visit recharging platforms more than once.

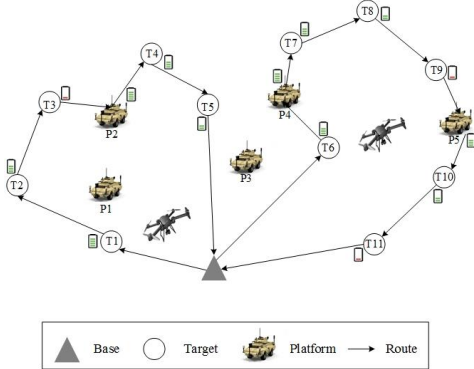


Fig. 1 An illustrative example of the UAV routing problem with recharging

Although the UAV routing problem with recharging is initially inspired from a military application, there are also many potential applications in civil area, e.g. information collection in wild areas, wild search & rescue, and agriculture monitoring etc. The main factors and constraints in problem are as follows:

1) UAVs

Small UAV is driven by the lithium battery whose capacity is limited. The battery consumption is mainly divided into three parts. Firstly, UAV consumes power during the flight between targets visited, and the speed of battery power consumption is related with the flying speed and distance. Secondly, when the UAV is collecting the information at a target, the sensors on the

UAV start to work and consume battery power. The consumption rate is related to the accuracy and duration of reconnaissance at the target. At other times, the sensor is turned off and does not consume battery power. However, the power consumption is still existed when the UAV is hovering above the target and waiting, which is the third part. In this paper, UAVs start from the base and must return to it after finishing all the reconnaissance missions within the specified time.

2) Recharging platform

Many ground vehicles in the battlefield are equipped with wireless charging devices, which can recharge the UAVs quickly. The recharging level of the UAV's battery is related to the recharging time. The location of these platforms and the base are known.

3) Targets

A set of targets are located at different positions in the area, and each target can only be detected in a specified time window. If the UAV arrives earlier than the earliest start time, the UAV needs to hover over the target and wait. The time windows and the location information of all the targets are given before the mission planning period.

The objective of the problem is to minimize the total mission time and the number of UAVs utilized through optimizing the flight routes for reconnaissance and recharging.

The notations applied in the following model formulation are summarized in TABLE I:

TABLE I
NOTATIONS USED IN THE FOLLOWING MODEL

Sets	
V_i	set of targets;
F	set of recharging platforms and their copies;
$\{0\}$	the starting base;
$\{n+1\}$	the ending base;
V^0	set of targets, recharging platforms and the starting base, $V^0 = V_i \cup F \cup \{0\}$;
V^{n+1}	set of targets, recharging platforms and ending base, $V^{n+1} = V_i \cup F \cup \{n+1\}$;
V	set of all vertices;
Parameters	
d_{ij}	the travel distance between node i and j ;
t_{ij}	the travel time between node i and j ;
g	the battery charge rate;
s_i	the reconnaissance time of target i ;
e_i	the earliest starting time of reconnaissance at target i ;
l_i	the latest starting time of reconnaissance at target i ;
Q	the capacity of the UAV's battery;
α	the weight coefficient of unit UAV;
β	the weight coefficient of total reconnaissance time, and $\alpha+\beta=1$;
c_d	the power consuming rate of UAV for travelling;
c_r	the power consuming rate of UAV for reconnaissance;
c_w	the power consuming rate of UAV for waiting;
c_{nav}	fixed cost of a UAV;

Decision variables

u_i	the starting time of reconnaissance target i , $i \in V_1$;
y_i	the remaining power level when the UAV arrives at the node i , $i \in V_1$;
q_i	the recharging quantity at recharging platform i , $i \in F$;
Y_i	the power level when the UAV leaves the recharging platform i , $i \in F$;
δ_i	the waiting time of the UAV at target i , $i \in V_1$;
x_{ij}	1, if a UAV travels from node i to node j ; otherwise 0;
z_i	1, if the recharging platform i is selected for recharging; otherwise 0;

The mathematical model of the UAV routing problem is formulated as follows:

$$\text{Min } \Pi = \alpha \cdot c_{uav} \sum_{j \in V_1 \cup F} x_{0j} + \beta \cdot \left\{ \sum_{i,j \in V} x_{ij} t_{ij} + \sum_{i \in F} z_i q_i g + \sum_{i \in V_1} \delta_i + \sum_{i \in V_1} s_i \right\} \quad (1)$$

Subject to

$$\sum_{j \in V^{n+1}, j \neq i} x_{ij} = 1, \quad \forall i \in V_1, \quad (2)$$

$$\sum_{j \in V^{n+1}, j \neq i} x_{ij} \leq 1, \quad \forall i \in F, \quad (3)$$

$$\sum_{i \in V^0, i \neq j} x_{ij} = \sum_{i \in V^{n+1}, i \neq j} x_{ji}, \quad \forall j \in V_1 \quad (4)$$

$$u_i + (t_{ij} + s_i) x_{ij} - l_0 (1 - x_{ij}) \leq u_j, \quad \forall i \in V^0, \forall j \in V^{n+1}, i \neq j, \quad (5)$$

$$u_i + t_{ij} x_{ij} + g q_i z_i - (l_0 + g Q) (1 - x_{ij}) \leq u_j, \quad \forall i \in F, \forall j \in V^{n+1}, i \neq j, \quad (6)$$

$$e_j \leq u_j \leq l_j, \quad \forall j \in V, \quad (7)$$

$$0 \leq y_j \leq y_i - (c_d \cdot d_{ij}) x_{ij} - c_r s_i x_{ij} - c_w \delta_i \cdot x_{ij} + Q(1 - x_{ij}), \quad \forall i \in V_1, \forall j \in V^{n+1}, i \neq j, \quad (8)$$

$$0 \leq y_j \leq Y_i - (c_d \cdot d_{ij}) x_{ij} + Q(1 - x_{ij}), \quad \forall i \in F, \forall j \in V^{n+1}, i \neq j, \quad (9)$$

$$Y_i = z_i (y_i + q_i), \quad \forall i \in F, \quad (10)$$

$$Y_i \leq Q, \quad \forall i \in F, \quad (11)$$

$$x_{ij} \in \{0,1\}, \quad \forall i \in V^0, \forall j \in V^{n+1}, i \neq j, \quad (12)$$

$$z_j \in \{0,1\}, \quad \forall j \in F, \quad (13)$$

$$q_i \geq 0, \quad \forall i \in F, \quad (14)$$

Objective (1) is to minimize a weighted sum of the total number of UAVs and the overall time for completing all targets' reconnaissance mission, where the weighted coefficients are determined by the planner. Constraint (2) ensures each target is visited only once, while constraint (3) handles the connectivity between the nodes (targets and recharging platforms). Constraint (4) guarantees that the number of outgoing arcs equals to the number of incoming arcs at each vertex. Constraints (5) and (6) enforce the feasibility of the time flow. Constraint (7) ensures the time windows of the targets and base. Constraints (8) and (9) balance the residual battery level after a visit to a target or a recharging platform and ensure that it is always non-negative. Constraint (10) determines the battery level after recharging at a recharging platform. Constraint (11) makes sure that the battery level at recharging platform is restricted to the UAV recharging capability. Constraints (12)-(14) define the decision variables.

In the model, the UAV's battery is allowed to be recharged partially, and thus only required battery power for the following route would be recharged so as to save time and energy.

IV. ADAPTIVE LARGE NEIGHBORHOOD SEARCH ALGORITHM

In this section, an improved adaptive large neighborhood search (ALNS) algorithm embedded with a recharging platform insertion heuristic is proposed. The ALNS framework was firstly proposed by Pisinger and Ropke [38-40], and has been widely used for solving the VRP [41-43] and EVRP [44-45]. The basic idea of ALNS is to employ different combinations of destroy and repair operators to obtain new neighborhood solutions while the utilizing probability of each operator is adaptively updated based on its weight which is related to its performance in the search process.

Algorithm 1: The main procedure of the improved ALNS algorithm with simulated annealing

Input: a set of destroy and repair operators (D and R), the cooling rate h

Output: the best solution Ψ_{best}

```

1 Using the two-stage constructive heuristic to construct an initial
  feasible solution  $\Psi_{current}$ ,  $\Psi_{best} \leftarrow \Psi_{current}$ ;
2 Initialize probability  $P_d^i$  for each destroy operator  $d \in D$  and
  probability  $P_r^i$  for each repair operator  $r \in R$  in the iteration round  $t$ ;
3 Let  $j$  be the counter initialized as  $j \leftarrow 1$  and  $T$  be the temperature;
4 while the maximum number of iterations is not reached do
5   Remove all the platforms in the current feasible solution  $\Psi_{current}$ ;
6   Select a destroy operator  $d^*$ , remove targets and obtain the
    solution  $\varphi_{new}^*$ ;
7   Select a repair operator  $r^*$ , insert the removed targets into the
    solution  $\varphi_{new}^*$  to generate the new NC solution  $\varphi_{new}$ ;
8   Using recharging platform insertion heuristic to fix the NC routes
    and construct the new feasible solution  $\Psi_{new}$ ;
9   if  $c(\Psi_{new}) < c(\Psi_{current})$  then
10     $\Psi_{current} \leftarrow \Psi_{new}$ ;
11  end
12 else
13   Let  $v \leftarrow e^{-(c(\Psi_{new}) - c(\Psi_{current}))/T}$ ;
14 end
15 Generate a random number  $\delta \in [0, 1]$ ;
16 if  $\delta < v$  then
17    $\Psi_{current} \leftarrow \Psi_{new}$ ;
18 end
19 if  $c(\Psi_{current}) < c(\Psi_{best})$  then
20    $\Psi_{best} \leftarrow \Psi_{current}$ ;
21 end
22  $T \leftarrow hT$  and let  $j \leftarrow j + 1$ ;
23 Update the probabilities of each operator;
24 end
25 return  $\Psi_{best}$ 

```

In this section, a two-stage constructive heuristic is used to generate an initial feasible solution for the problem. First, the nearest neighborhood heuristic is utilized to construct a solution without regarding the battery capacity constraint. In this stage, each UAV's route satisfies the constraints on time window, while the battery capacity may be not enough to support the UAV to complete the flight of the route. Thus, in the second stage, a recharging platform insertion heuristic is designed to optimize where to recharge the UAV and how much energy

should be recharged, so as to make the route feasible. The detail process of the insertion heuristic is described in the following Subsection A.

In the search process of ALNS, any change on the targets in a route could affect the recharging decisions. In order to improve the search efficiency, the destroy and repair operations in ALNS are conducted on the Non-Charged (NC) solution where all the recharging platforms are removed from the UAV's routes. When a NC solution is generated after the destroy and repair operations, the recharging platform insertion heuristic is employed to make it feasible. Thus, each neighborhood search operation follows a rule of 'Search Firstly, and Insertion Afterwards'. The new feasible solution is accepted with the SA criterion and the main procedure of the ALNS is shown in Algorithm 1.

A. Recharging platform insertion heuristic

Due to the battery capacity, the endurance range of UAV is limited. If the total energy consumption of UAV exceeds its battery capacity, it has to visit a recharging platform for recharging so as to visit the following targets. A recharging platform insertion heuristic is proposed to optimize the recharging decisions on where and how much to recharge. The main steps of the recharging platform insertion heuristic are shown in Algorithm 2.

Algorithm 2: The procedure of the recharging platform insertion heuristic

Input: An infeasible NC solution φ_0
Output: A feasible solution Ψ_{fea}

```

1 Let  $V^{unvisited} \leftarrow \Phi$  and find all the infeasible routes  $\gamma_{in} \in \varphi_0$ ;
2 for  $\gamma_{in} \in \varphi_0$  do
3   Find all the feasible insertion places and denoted as a set  $\Pi_{inset}$ ;
4   Let  $i$  be the counter and initialized as  $i \leftarrow 1$ ;
5   for  $i \in \Pi_{inset}$  do
6     Insert the nearest accessible recharging platform in the
       corresponding places;
7     Determine the recharging level  $q_i$ ;
8     Check the time windows;
9     if the time windows are violated then
10      Remove the time-violated target from the route and obtain
        the current feasible route  $\gamma_{cur}$ ;
11      Add the removed targets into the set  $V^{unvisited}_{cur}$ ;
12    end
13    if the route  $\gamma_{cur}$  satisfies the criterion of acceptance then
14       $\gamma_{in} \leftarrow \gamma_{cur}$ ;
15      Add the removed targets in  $V^{unvisited}_{cur}$  into  $V^{unvisited}$ ;
16    end
17     $i \leftarrow i + 1$ ;
18  end
19 end
20 while  $V^{unvisited} \neq \Phi$  do
21   Utilize the nearest neighborhood heuristic to form NC routes for
     targets in  $V^{unvisited}$ ;
22   Repeat line 1-19;
23 end
24 return  $\Psi_{fea}$ 

```

In Algorithm 2, the insertion position of recharging platform is firstly optimized. For an NC route where the total travel

distance violates the maximum battery capacity, the UAV starts from the base and travels along the route until the target that cannot be visited within the left battery level. Then find the nearest recharging platform which it could reach by the current battery power, and insert it after the current target. To find the best inserting position of the recharging platform, all possible insertions between adjacent targets are compared. Considering the two routes (a) and (b) in Fig. 2 as an example, route (a) and route (b) are the two different feasible routes after inserting recharging platforms into the same NC route at different insertion places. Although UAV in route (b) need to be recharged twice, the total time is a little shorter than that of route (a) due to the distribution of recharging platforms. In this condition, we would choose route (b) instead of (a).

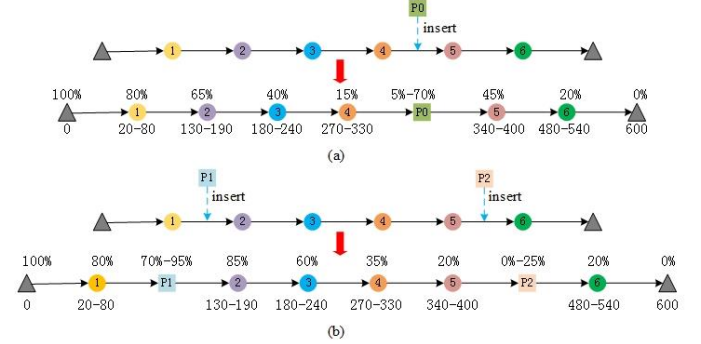


Fig. 2. An illustration for different insertion places of recharging platforms

After the insertion position of the recharging platform is determined, the UAV can be partially recharged and the recharging level is determined by the required battery power of the subsequent route. After the recharging level at each recharging platform is known, the time windows of the targets after the recharging platform are influenced and the feasibility should be checked again. If there are targets whose time windows are not satisfied, they should be removed into the list of $V^{unvisited}$. As shown in Fig. 3, after inserting P4 between target 3 and target 4, the time windows of target 4 is violated due to the long recharging time at P4. Thus, target 4 needs to be removed from the route. For the targets in $V^{unvisited}$, we iteratively use the two-stage constructive heuristic until a feasible solution is generated. Finally, many feasible routes will be generated based on the same NC route after inserting recharging platforms in different positions, and the best route will be accepted which keeps more targets and consumes shorter time for completing the route in the second place.

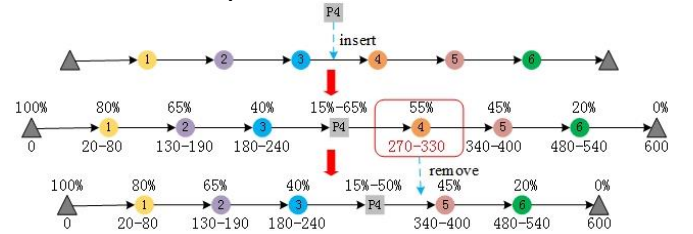


Fig. 3. Check the time windows

B. Neighborhood Structures

Given an initial feasible solution, remove all the platforms and generate a NC solution. The neighborhood search is applied through different destroy and repair operators on the NC

solution. Firstly, a removal operator is selected and a certain number of targets are removed based on the corresponding removal rule. Then, an insertion operator is conducted to repair the destroyed NC solution through inserting the removed targets into the current solution. Finally, the recharging platform insertion heuristic is used to insert the recharging platforms in the NC solution and generate a feasible solution.

1) Removal operators

The destroy mechanism of the proposed ALNS framework consists of different removal operators. And nine removal operators are adapted, which are grouped into two types: Route Removal (RR) and Targets Removal (TR). In RR, a UAV route is selected and removed. In TR, a subset of λ targets are selected and moved into the removal list ς . The value of parameter λ is determined according to the overall number of targets n_c , and here it is generated between n_c and \bar{n}_c following the uniform distribution. All removal operators are introduced below.

Random Route Removal A route is randomly chosen from the set of all routes, in which all the targets are removed and added into the removal list ς . The randomly choosing strategy can extend the search range in the solution space and increase the probability of finding the global optimum.

Shortest Route Removal This operator chooses the shortest route in the current solution and removes all the targets in this selected route. The purpose of this operator is to maximize the utility of the UAV and reduce the number of UAVs.

Random Target Removal A set of λ targets are randomly removed from different routes in the current solution. The goal of random selection is also to diversify and enlarge the search range in the solution space.

Worst-Distance Target Removal The worst distance targets are selected one by one and removed from the current solution. Here the distance of a target means a sum of the distance from the current target to its preceding target and the distance from the current target to its succeeding target in the route.

Worst-Time Target Removal This operator calculates, for each target i , the difference between reconnaissance start time and the corresponding earliest start time e_i , and then iteratively removes the target with the largest difference. The purpose is to avoid long wait or delayed reconnaissance starting time in order to satisfying the time windows of more targets to the maximum extent.

Modified Shaw Removal The idea of this operator is to remove a set of targets according to a specified rule. This operator is adapted from the Shaw Removal which was proposed by Shaw (1998). The difference of general rule between the modified and the origin Shaw Removal is that UAVs routing do not consider the load capacity but consider the reconnaissance time since the reconnaissance duration effect the power consumption. Thus, operator starts by removing a node i randomly. Let $l_{ij} = -1$ if the target i and target j are both detected by the same UAV in the same route, otherwise $l_{ij} = 1$ and selects a node $j^* = \arg \min \{ \Phi_1 d_{ij} + \Phi_2 |e_i - e_j| + \Phi_3 l_{ij} + \Phi_4 |s_i - s_j| \}$, where $\Phi_1 - \Phi_4$ are the weights parameters.

Proximity-based Target Removal In this operator, the first target is randomly selected, and then the nearest target to the former selected one is selected. In each selection, we always select the target nearest to the former one. According to this strategy, a set of λ targets are selected one by one and removed. The working way of the operator is illustrated in Fig. 4.

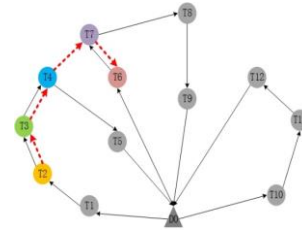


Fig. 4. Proximity-based target removal

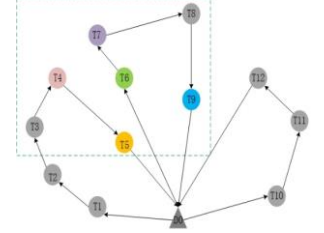


Fig. 5. Zone removal

Time-based Target Removal Similar to the Proximity-based Target Removal, this operator selects a number of targets with proximate time windows and removes them from the current solution.

Zone Removal The operator firstly randomly defines an area with a predefined size in the Cartesian coordinate system and randomly selects λ targets to remove which are located in the area. If there are less than λ targets in the selected area, then reselects an area with the same size, and continue the same procedure until λ nodes are removed. As Fig. 5 presents, the green rectangular box represents the selected area and then randomly selects target T4, T5, T6, T7 and T9 in the area to remove. The pseudocode of zone removal is given in Algorithm 3.

Algorithm 3: The pseudocode of zone removal

Input: An infeasible NC solution φ_0 , numbers of targets to be removed λ

Output: A destroyed solution φ_1 , removal list to be repaired ς

```

1 Let  $\varsigma \leftarrow \Phi$  and initialize number of removed targets current as  $\lambda_{cur} \leftarrow \lambda$ ;
2 Randomly define an area and calculate the number of targets  $m$  in the area;
3 if  $m < \lambda_{cur}$  then
4   Remove all the  $m$  targets in the area from its routes and add the removed targets into list  $\varsigma$ ;
5 end
6 else
7   Randomly remove  $\lambda_{cur}$  targets in the area from its routes and add the removed targets into list  $\varsigma$ ;
8 end
9 while the number of removed targets  $\eta$  in the list  $\varsigma$  is less than  $\lambda$  do
10   Let  $\lambda_{cur} \leftarrow (\lambda - \eta)$ ;
11   Repeat line 2-8;
12 end
13 return  $\varphi_1, \varsigma$ 

```

2) Insertion operator

Five insertion operators are introduced, which are used for inserting the removed targets back into the routes to generate a new NC solution. In the inserting process, the constraints on

time windows must be satisfied, while the battery capacity is not considered.

Greedy Insertion The operator calculates the insertion cost of each removed target in the best insertion position, and repeatedly inserts the node with the least insertion cost in its best feasible position of the current solution. The insertion cost here is calculated as the increased distance.

Regret-2 Insertion Let Δf_i represent the difference in the objective function value after inserting target i in the current solution. Let $i^* = \arg \max_{i \in \mathcal{C}} \{\Delta f_{i2} - \Delta f_{i1}\}$, where Δf_{i1} is the difference after inserting target i in the best feasible insertion place and Δf_{i2} is the difference after inserting target i in the second-best insertion place.

Regret-3 Insertion Similar to the Regret-2 Insertion operator, Δf_i represents the same meaning. Let $i^* = \arg \max_{i \in \mathcal{C}} \{\Delta f_{i3} - \Delta f_{i1}\}$, where Δf_{i1} is the difference after inserting the target i in the best feasible insertion place and Δf_{i3} is the difference after inserting the target i in the third-best insertion place.

Time-based Insertion The idea of this operator originates from the Greedy Insertion, the unique difference between the two operator is the calculation of the insertion cost. The operator defines the insertion cost as the change in the finish time of the route.

Zone Insertion The operator selects the removed targets by the criterion of the Time-based Insertion operator above. The unique difference is that it only considers the routes in a specific zone, instead of investigating all routes in the current destroyed solution.

C. Adaptive adjustment of the operators' weight

The selection of the removal and insertion operators is governed by a roulette-wheel mechanism. If we have k operators with weights w_i , $i \in \{1, 2, \dots, k\}$ and we choose the operator j with probability

$$p_j^{\text{select}} = \frac{w_j}{\sum_{i=1}^k w_i} \quad (15)$$

At the beginning, all removal or insertion operators have the same probability. Thus, in this paper, there are nine removal operators and five insertion operators, the initial weight of each removal operator and insertion operator is set to 1/9 and 1/5, respectively. During the searching process, they are updates as follows:

$$w_i^{t+1} = w_i^t(1-r) + r \cdot \pi_i / \omega_i \quad (16)$$

where w_i^{t+1} is the weight of operator i in iteration round $t+1$, r is the roulette-wheel parameter, π_i is the score of the operator i and ω_i is the number of times it was used during the last N_w iterations. If a new global best solution is obtained, the score π_i of the operator i is increased by σ_1 . If the new feasible solution is not a global best solution but better than the current solution, the score π_i is increased by σ_2 . If the solution is worse than the current solution but can be accepted within a certain probability, the score π_i is increased by σ_3 . Thus, the

score π_i of operator i is determined by the sum of σ_1 , σ_2 , σ_3 obtained in each iteration.

D. Acceptance and stopping criteria

The simulated annealing strategy is adopted as an acceptance criterion in the framework of ALNS algorithm. During the searching process, Ψ_{best} is the global best solution, Ψ_{current} is the current solution before the iteration begins, and Ψ_{new} is the new feasible solution after the iteration. Let $c(\Psi)$ denotes the objective function value of solution Ψ , a solution Ψ_{new} is always accepted if $c(\Psi_{\text{new}}) < c(\Psi_{\text{current}})$, and accepted with probability $e^{-(c(\Psi_{\text{new}}) - c(\Psi_{\text{current}}))/T}$. If $c(\Psi_{\text{new}}) > c(\Psi_{\text{current}})$, where T denotes the temperature. In the iteration process, the temperature is gradually decreased at a constant rate of hT , where $0 < h < 1$ is a constant parameter. The algorithm returns the global best solution after the maximum number of iterations.

V. COMPUTATIONAL EXPERIMENTS

As it is the first work to investigate UAV routing problem with recharging, there is no benchmark dataset exists. To analyze the performance of the proposed ALNS algorithm, we designed a new set of benchmark instances based on the well-known Solomon dataset. All the algorithms were programmed with Visual C++, and all experiments are conducted on a laptop with Intel Core i5 processor (3GHz) and 8GB RAM. The data set in the following subsections are presented in the supplement[53].

A. Experiment design

A set of 56 large instances are designed, and there are 100 targets and 21 recharging platforms in each instance. The distribution of targets and recharging platforms is referred the data set in [12], which is designed based on the Solomon dataset. These instances can be divided into 3 classes according to the geographical distribution of targets: Random distribution (R), clustered distribution (C) and a mixture of both (RC). Further, the instances with narrow time windows are classified in group R1, C1 and RC1, while the ones with wide time windows are classified in group R2, C2 and RC2. The detail information for all nodes in the 56 instances can be found in the supplement.

The battery capacity of UAV is a constant and is set as 150. Depending on different conditions, the battery power consumption rate is set to 1 when UAV is flying en-route, and is set to 0.5 when it is hovering above targets and waiting. When the UAV is collecting information on a target, it consumes more battery power than any time during the travelling and the energy consumption rate is set to 2. Besides, the average velocity of UAV is set to 1 and the inverse recharging rate is set to 0.33 which means that a complete recharging from zero battery level requires 50 minutes.

B. Algorithm performance

To analyze the performance of ALNS, it is compared with two widely used metaheuristics for routing problems, which are the ant colony optimization (ACO) algorithm and the variable neighborhood search (VNS) algorithm. ACO is firstly proposed by Dorigo et al. [46] and were widely used to solve the VRP and EVRP [47-49]. VNS performs local search on large

neighborhoods and its effectiveness is also verified by former works [12, 50-51]. In this experiment, we applied the framework of ACO and traditional VNS from the work [52].

All the 56 instances are solved by ALNS, ACO and VNS respectively. In TABLE II-IV, the objective function value of best solution in 10 runs are reported. Furthermore, we calculate the relative gaps between ALNS and ACO ($\Delta_1\%$) for the objective function value, where $\Delta_1\% = (Obj.ACO - Obj.ALNS) / Obj.ALNS$, and the gaps between ALNS and VNS ($\Delta_2\%$), where $\Delta_2\% = (Obj.VNS - Obj.ALNS) / Obj.ALNS$. Thus, a negative value of $\Delta_1\% / \Delta_2\%$ means relative improvement obtained by ALNS.

TABLE II
THE RESULTS OF R TYPE INSTANCES UNDER DIFFERENT METHODS

Inst.	Objective			$\Delta_1\%$	$\Delta_2\%$
	ALNS	ACO	VNS		
R101	2268.48	2956.03	2926.82	-23.59	-22.49
R102	2027.17	2755.44	2631.63	-26.43	-22.97
R103	1780.75	2302.98	2318.19	-22.68	-23.18
R104	1557.14	1969.71	1916.42	-20.95	-18.75
R105	1823.43	2318.40	2250.37	-21.35	-18.97
R106	1789.65	2216.38	2292.68	-19.25	-21.94
R107	1663.56	2184.44	2143.02	-23.85	-22.37
R108	1580.92	1793.02	1795.81	-11.83	-11.97
R109	1745.17	2115.57	2132.72	-17.51	-18.17
R110	1559.72	1949.06	1863.35	-19.98	-16.30
R111	1598.49	2010.06	1933.80	-20.48	-17.34
R112	1522.88	1687.59	1599.38	-9.76	-4.78
Average				-19.81	-18.27
R201	864.59	1155.83	1296.84	-25.20	-33.33
R202	781.24	1209.12	1272.52	-35.39	-38.61
R203	741.97	1107.75	1121.97	-33.02	-33.87
R204	702.30	906.33	985.42	-22.51	-28.73
R205	800.17	968.27	1003.73	-17.36	-20.28
R206	757.01	1108.67	1004.70	-31.72	-24.65
R207	715.33	942.56	968.66	-24.11	-26.15
R208	706.93	837.72	844.08	-15.61	-16.25
R209	750.02	981.09	929.40	-23.55	-19.30
R210	749.06	1085.46	976.31	-30.99	-23.28
R211	755.92	848.93	844.52	-10.96	-10.49
Average				-24.58	-24.99

TABLE III
THE RESULTS OF C TYPE INSTANCES UNDER DIFFERENT METHODS

Inst.	Objective			$\Delta_1\%$	$\Delta_2\%$
	ALNS	ACO	VNS		
C101	1402.02	1621.00	1746.16	-13.51	-19.71
C102	1350.14	1844.74	1675.53	-26.81	-19.42
C103	1277.20	1744.82	1670.86	-26.80	-23.56
C104	1192.49	1450.37	1306.90	-17.78	-8.75
C105	1275.09	1415.71	1524.82	-9.93	-16.38
C106	1262.51	1413.80	1408.24	-10.70	-10.35
C107	1196.36	1357.22	1423.80	-11.85	-15.97
C108	1198.96	1267.97	1404.42	-5.44	-14.63
C109	1138.29	1219.00	1334.79	-6.62	-14.72
Average				-14.38	-15.94
C201	381.28	468.33	602.79	-18.59	-36.75
C202	382.64	472.10	548.81	-18.95	-30.28
C203	377.88	483.49	563.14	-21.84	-32.90
C204	360.83	397.35	437.16	-9.19	-17.46
C205	371.66	388.31	484.85	-4.29	-23.35
C206	363.27	386.11	401.39	-5.92	-9.50
C207	365.47	393.93	468.23	-7.22	-21.95
C208	365.39	389.16	398.28	-6.11	-8.26
Average				-11.51	-22.56

The 23 R type instances can be further divided into two classes, noted as R1 and R2. Targets in R1 instances have narrow time windows, while targets in R2 instances have wide time windows. The computational results in TABLE II show that the ALNS algorithm can achieve better solutions than other algorithms for all the R instances. Furthermore, the value of objective function for all R1 instances are reduced by an average of 19.81% to ACO and an average of 18.27% to VNS, while the value of objective function for all R2 instances the instances are reduced by an average of 24.58% to ACO and an average of 24.99% to VNS.

There are also two classes of instances in the C type and RC type data set respectively, which are noted as C1/RC1 and C2/RC2. TABLE III reports all the computational results of C type instances, and TABLE IV reports the results of RC type instances. It can be found that the ALNS algorithm performs better than ACO and VNS for all C and RC instances. Generally speaking, the results for C2 and RC2 instances are also better compared to those of C1 and RC1 respectively. Therefore, it shows that the ALNS algorithm works better for instances with wide time window, and the ALNS algorithm can efficiently enlarge the search space in the situation with loose constraints on time window.

TABLE IV
THE RESULTS OF RC TYPE INSTANCES UNDER DIFFERENT METHODS

Inst.	Objective			$\Delta_1\%$	$\Delta_2\%$
	ALNS	ACO	VNS		
RC101	2333.83	2856.69	2869.70	-18.30	-18.67
RC102	2233.72	2778.83	2662.54	-19.62	-16.11
RC103	2055.01	2507.59	2396.11	-18.05	-14.24
RC104	1787.31	2032.21	1969.86	-12.05	-9.27
RC105	2185.81	2605.71	2594.42	-16.12	-15.75
RC106	2047.81	2400.05	2372.10	-14.68	-13.67
RC107	1861.90	2132.97	2019.12	-12.71	-7.79
RC108	1734.64	1970.73	1787.89	-11.98	-2.98
Average				-15.44	-12.31
RC201	557.43	865.43	788.91	-35.59	-29.34
RC202	475.83	827.74	789.15	-42.52	-39.70
RC203	430.69	679.95	736.78	-36.66	-41.54
RC204	379.19	500.88	558.89	-24.30	-32.15
RC205	459.21	722.68	739.63	-36.46	-37.91
RC206	429.98	601.60	598.37	-28.53	-28.14
RC207	400.72	580.31	560.59	-30.95	-28.52
RC208	338.62	449.39	419.52	-24.65	-19.28
Average				-32.46	-32.07

The results verify that the performance of ALNS has significant superiority for solving the URP-RC problem. The ALNS obtained the best solution in all 56 instances. As for the fixed cost of vehicles, the ALNS always reached the minimum which contributes to the best solution in a great measure.

C. Analysis on the impact of the recharging strategy

An important characteristic of the investigated problem is allowing the UAV to be recharged in the route, which is expected to expand the endurance range of UAV and improve the routing efficiency with lower cost. To analyze the impact of recharging strategy, the ALNS algorithm is used to solve all the instances with recharging (URP-RC) and without recharging (URP). We analyzed the number of UAVs (m), total mission completing time (TT) and the objective function value (Obj)

respectively. ALNS is run 10 times for each instance, and the best solutions for all instances are reported in TABLE V-VII.

From the computational results, it can be seen that the number of UAVs for URP-RC is much less than that for URP. For the R1 and RC1 type instances, the total completing time for URP-RC is also less than that for URP, while for the other instances, the URP mode can complete the task in much less time. That is because the UAVs in URP-RC have to visit multiple recharging platforms in the routes for most of the situations, which consume much additional time. Usually, the UAVs are critical resources in both military and civil application, and its cost are relatively higher than the routing cost. The number of UAVs is greatly reduced in URP-RC which

TABLE V
THE RESULTS OF R TYPE INSTANCES BETWEEN URP-RC AND URP

Inst.	URP-RC			URP		
	m	TT	Obj	m	TT	Obj
R101	20	2757.78	2378.89	35	2831.96	3165.98
R102	19	2364.90	2132.45	32	2461.78	2830.89
R103	17	2139.18	1919.59	30	2146.56	2573.28
R104	16	1877.20	1738.60	29	1934.60	2417.30
R105	17	2038.62	1869.31	31	2276.36	2688.18
R106	17	1931.54	1815.77	29	2142.44	2521.22
R107	16	1969.50	1784.75	29	2035.20	2467.60
R108	15	1921.74	1710.87	28	1919.48	2359.74
R109	16	1940.00	1770.00	29	2044.30	2472.15
R110	15	1863.86	1681.93	28	1937.58	2368.79
R111	16	1916.68	1758.34	29	1984.68	2442.34
R112	15	1677.68	1588.84	28	1869.66	2334.83
R201	9	904.74	902.37	19	663.34	1281.67
R202	9	821.40	860.70	18	469.38	1134.69
R203	8	736.68	768.34	18	383.98	1091.99
R204	8	671.36	735.68	17	370.60	1035.30
R205	9	725.58	812.79	17	393.14	1046.57
R206	8	714.76	757.38	18	374.66	1087.33
R207	8	678.68	739.34	17	352.08	1026.04
R208	8	637.84	718.92	17	326.10	1013.05
R209	8	700.04	750.02	17	346.58	1023.29
R210	9	757.94	828.97	17	363.70	1031.85
R211	8	711.84	755.92	17	347.08	1023.54

TABLE VI
THE RESULTS OF C TYPE INSTANCES BETWEEN URP-RC AND URP

Inst.	URP-RC			URP		
	m	TT	Obj	m	TT	Obj
C101	16	1265.20	1432.60	34	590.26	1995.13
C102	16	1185.14	1392.57	34	498.62	1949.31
C103	15	1196.60	1348.30	33	469.40	1884.70
C104	15	1032.52	1266.26	33	460.42	1880.21
C105	15	1089.52	1294.76	34	504.06	1952.03
C106	16	1049.12	1324.56	34	509.76	1954.88
C107	15	1100.66	1300.33	34	468.62	1934.31
C108	14	1013.96	1206.98	34	481.62	1940.81
C109	14	1040.26	1220.13	33	460.00	1880.00
C201	6	285.66	442.83	10	126.58	563.29
C202	6	233.06	416.53	10	97.76	548.88
C203	5	255.76	377.88	10	78.18	539.09
C204	5	257.16	378.58	10	77.50	538.75
C205	5	271.46	385.73	10	91.24	545.62
C206	5	241.04	370.52	10	70.80	535.40

C207	5	247.16	373.58	10	74.36	537.18
C208	5	241.82	370.91	9	71.72	485.86

causes the objective function value is smaller in general. And the results of RC type instances in TABLE VII shows that URP-RC performs better for each aspect of the objective than URP.

From TABLE V-VII, we can see that URP-RC obtained better solution with much lesser number of UAVs and smaller objective function values for all the instances compared to URP. Therefore, the recharging strategy does improve the efficiency and reduce the cost of UAVs.

D. Sensitivity analysis on the battery capacity

The battery capacity plays a key role in the UAV routing problem. In order to analyze the impact of different battery capacities, we vary the value of battery capacity from 120 to 300 while the value of other parameters keep unchanged. The number of UAVs (m) and the overall mission time (TT) are calculated under different battery capacities respectively by ALNS. Also, we compared the impact of battery capacities between URP-RC and URP. The results for R101 instance are reported in Fig. 6. The experimental results on other instances are similar, and the parallel analysis on the other instances are omitted here.

TABLE VII
THE RESULTS OF RC TYPE INSTANCES BETWEEN URP-RC AND URP

Inst.	URP-RC			URP		
	m	TT	Obj	m	TT	Obj
RC101	20	3086.26	2801.74	41	3646.58	3873.29
RC102	19	2836.84	2559.44	39	3410.76	3655.38
RC103	18	2492.08	2536.54	37	3171.60	3435.80
RC104	15	2203.28	2231.74	37	3075.14	3387.57
RC105	20	2675.84	2665.50	38	3318.66	3559.33
RC106	19	2527.52	2527.76	38	3259.80	3529.90
RC107	16	2318.62	2150.62	37	3171.94	3435.97
RC108	15	2154.04	2056.74	37	3100.46	3400.23
RC201	5	755.06	660.02	8	586.24	693.12
RC202	4	635.46	581.48	8	463.18	631.59
RC203	4	501.52	507.82	8	405.58	602.79
RC204	4	397.42	363.34	7	302.74	501.37
RC205	5	554.12	517.18	8	420.44	610.22
RC206	4	528.28	484.68	7	410.08	555.04
RC207	4	467.96	456.04	7	300.22	500.11
RC208	4	383.68	338.62	6	248.86	424.43

From Fig. 6(a), it can be observed that the number of UAVs decreases sharply with the increase of battery capacity in URP and comparatively decreases gently in URP-RC. When the battery capacity is 120, the deviation of the number of UAVs between URP-RC and URP reaches 30 which verifies that URP-RC can significantly reduce the fixed cost of UAVs especially for the small UAVs with short endurance range. With the increase of battery capacity, the deviation between URP-RC and URP becomes small and reaches 0 when the capacity is 300. That is because the UAVs can complete the task without recharging when the battery capacity is large enough. Similarly, the results in Fig.6(b) showed that the overall mission time decreases as the battery capacity increases, and becomes the same for both URP-RC and URP when the battery capacity is large enough. Since the targets have specific time windows in

both URP-RC and URP, when the UAV arrives earlier than the earliest reconnaissance start time, it has to wait for some time. Therefore, the relatively deviation of the overall mission time between URP-RC and URP is smaller than that for the number of UAVs in Fig. 6(a).

The results in Fig. 6 confirm that the battery capacity does have an important impact on the routing of UAVs, and indicate that recharging is efficient to improve small UAVs' routing performance when the battery capacity is not large enough.

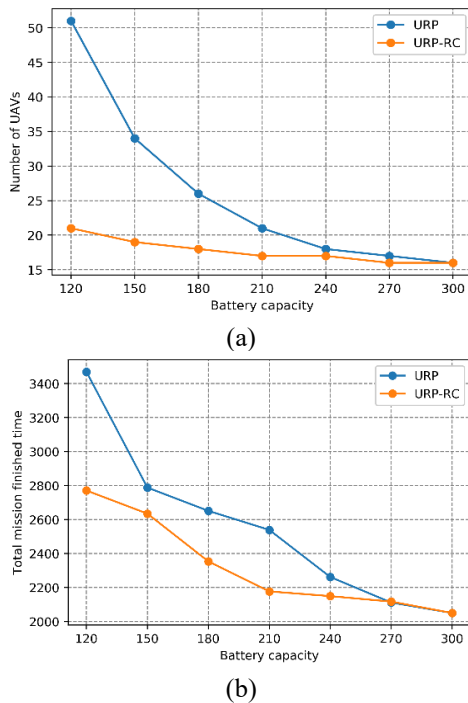


Fig. 6. The results for R101 instance under different battery capacities. (a) The number of UAVs under different battery capacities. (b) Total mission time under different battery capacities.

VI. CONCLUSION

To promote the utilization of small UAVs, we investigated a UAV routing problem with recharging to extend the endurance range of small UAVs, where the recharging platforms are fixed and UAV can fly to these recharging platforms to recharge its battery in the route. To solve this problem, a mixed integer nonlinear programming model is established and an improved ALNS algorithm embedded with a recharging platform insertion heuristic is designed. Furthermore, we create a set of benchmark instances based on the well-known Solomon dataset and the experiment results showed that the proposed ALNS performs significantly better than the ACO and VNS algorithm. Besides, the advantage of recharging strategy is verified and the sensitivity analysis on the battery capacity shows that the battery capacity effect both the number of UAVs and the objective values in URP-RC and URP.

To extend the proposed study, future research directions can be taken into consideration for the problem in a dynamic environment where the recharging platforms are mobile and has time windows as well. In many practical applications, the recharging platforms are mobile, such as public transport, which has recharging capacity and time window. Thus, it is a

meaningful and valuable extension in the model development. Another meaningful extension is studying new algorithms to solve this problem more efficiently and quickly.

REFERENCES

- [1] Otto A, Agatz N, Campbell J, et al. Optimization approaches for civil applications of unmanned aerial vehicles (UAVs) or aerial drones: A survey[J]. *Networks*, 2018, 72(4):411-458.
- [2] Kastrenakes, J. (2017). UPS has a delivery truck that can launch a drone. <http://www.theverge.com/2017/2/21/14691062/ups-drone-delivery-truck-test-completed-video/>. [Online; accessed 19-July-2018].
- [3] Aufgebauer, K. (2016). The future has landed, thanks to delivery drones. <http://goglobal.dhl-usa.com/blog/shipping/the-future-has-landed-thanks-to-delivery-drones/>. [Online; accessed 20-July-2018].
- [4] Lin, J. (2018). Meet China's growing fleet of automated delivery drones. <http://www.popsci.com/china-drone-deliveries/>. [Online; accessed 19-August-2018].
- [5] Johnson, L. (2017). 9 things you need to know about the Amazon Prime Air drone delivery service. <http://www.digitalspy.com/tech/feature/a820748/amazon-prime-air-drone-delivery-service/>. [Online; last accessed 18-Nov-2020].
- [6] Yao Liu, Zhihao Luo, Zhong Liu, Jianmai Shi, Guangquan Chen. Cooperative Routing Problem for Ground Vehicle and Unmanned Aerial Vehicle: The Application on Intelligence, Surveillance and Reconnaissance Missions[J]. *IEEE Access*, 2019, 7 (1): 63504-63518.
- [7] Scott Morton, Ruben D'Sa, Nikolaos Papanikolopoulos. Solar powered UAV: Design and experiments[C]// IEEE/RSJ International Conference on Intelligent Robots & Systems. IEEE, 2015.
- [8] Toksoz T, Redding J, Michini M, et al. Automated Battery Swap and Recharge to Enable Persistent UAV Missions[C]// AIAA Infotech@Aerospace Conference. 2011.
- [9] Hosseini S, Dai R, Mesbahi M. Optimal path planning and power allocation for a long endurance solar-powered UAV[C]// American Control Conference (ACC), IEEE, 2013, pp:2588-2593.
- [10] Illhove H, Young J J, Young D K, et al. System Optimization for Dynamic Wireless Charging Electric Vehicles Operating in a Multiple-Route Environment[J]. *IEEE Transactions on Intelligent Transportation Systems*, 2018, 19(6): 1709-1726.
- [11] Erdelić, Tomislav; Carić, Tonči. A Survey on the Electric Vehicle Routing Problem: Variants and Solution Approaches[J]. *Journal of Advanced Transportation*, 2019(54):1-48.
- [12] Schneider, M., Stenger, A., & Goeke, D. The electric vehicle-routing problem with time windows and recharging stations[J]. *Transportation Science*, 48(4) (2014) 500-520.
- [13] Shetty V K, Sudit M, Nagi R. Priority-based assignment and routing of a fleet of unmanned combat aerial vehicles [J]. *Computers & Operations Research*, 2008, 35 (6): 1813-1828.
- [14] Ceccarelli N, Enright J J, Frazzoli E, et al. Micro UAV Path Planning for Reconnaissance in Wind[C]// American Control Conference. IEEE, 2007.
- [15] Mufalli F, Batta R, Nagi R. Simultaneous sensor selection and routing of unmanned aerial vehicles for complex mission plans [J]. *Computers & Operations Research*, 2012, 39 (11): 2787-2799.
- [16] Avellar Gustavo, Pereira Guilherme, Pimenta Luciano et. al. Multi-UAV Routing for Area Coverage and Remote Sensing with Minimum Time[J]. *Sensors*, 2015, 15(11):27783-27803.
- [17] Evers L, Dollevoet T, Barros A I, Monsuur H. Robust UAV mission planning [J]. *Annals of Operations Research*, 2014, 222 (1): 293-315.
- [18] Mahmud I, Cho Y Z. Detection Avoidance and Priority-Aware Target Tracking for UAV Group Reconnaissance Operations[J]. *Journal of intelligent & robotic systems: Theory & applications*, 2018, 92: pp. 381-392.
- [19] Vanegas, F.; Gonzalez, F. Enabling uav navigation with sensor and environmental uncertainty in cluttered and gps-denied environments[J]. *Sensors* 2016, 16(5):666.
- [20] Zhihao Luo, Zhong Liu, Jianmai Shi. A two-echelon cooperated routing problem for a ground vehicle and its carried unmanned aerial vehicle[J]. *Sensors*, 2017, 17(5):1144.
- [21] J.E. Scott and C.H. Scott. Models for drone delivery of medications and other healthcare items. *International Journal of Healthcare Information Systems and Informatics (IJHISI)*, 2018, 13(3):20-34.
- [22] Zhou H, Kong H, Wei L, et al. Efficient Road Detection and Tracking for Unmanned Aerial Vehicle[J]. *IEEE Transactions on Intelligent Transportation Systems*, 2015, 16(1):297-309.

- [23] B. Rabta, C. Wankmuller, and G. Reiner. A drone fleet model for last-mile distribution in disaster relief operations[J]. *International Journal of Disaster Risk Reduction*, 2018, 28:107-112.
- [24] K. Dorling, J. Heinrichs, G. G. Messier and S. Magierowski, "Vehicle Routing Problems for Drone Delivery[J]," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 47, no. 1, pp. 70-85, Jan. 2017. DOI: 10.1109/TSMC.2016.2582745
- [25] Sawaditang S, Niyato D, Tan P S, et al. Joint Ground and Aerial Package Delivery Services: A Stochastic Optimization Approach[J]. *IEEE Transactions on Intelligent Transportation Systems*, 2018, PP:1-14.
- [26] Pham H X, La H M, Feil-Seifer D, et al. A Distributed Control Framework of Multiple Unmanned Aerial Vehicles for Dynamic Wildfire Tracking[J]. *IEEE Transactions on Systems Man & Cybernetics Systems*, 2018:1-12.
- [27] Chiang W C, Li Y, Shang J, et al. Impact of drone delivery on sustainability and cost: Realizing the UAV potential through vehicle routing optimization[J]. *Applied Energy*, 2019, 242(P1-1284):1164-1175.
- [28] S. Kim and I. Moon, "Traveling Salesman Problem With a Drone Station[J]," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 49, no. 1, pp. 42-52, Jan. 2019. DOI: 10.1109/TSMC.2018.2867496
- [29] Yao. Liu, Zhong. Liu, Jinamai. Shi, G. Wu and W. Pedrycz, "Two-Echelon Routing Problem for Parcel Delivery by Cooperated Truck and Drone[J]," in *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, doi: 10.1109/TSMC.2020.2968839.
- [30] Coelho B N, Coelho V N, Coelho I M, et al. A multi-objective green UAV routing problem[J]. *Computers & Operations Research*, 2017,88:306-315.
- [31] Li B, Patankar S, Moridian B, et al., "Planning Large-Scale Search and Rescue using Team of UAVs and Charging Stations*", 2018 *IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, Philadelphia, PA, 2018, pp. 1-8.
- [32] Ribeiro R G, Junior J R C, Cota L P, et al. Unmanned Aerial Vehicle Location Routing Problem with Charging Stations for Belt Conveyor Inspection System in the Mining Industry[J]. *IEEE Transactions on Intelligent Transportation Systems*, 2019, PP (99):1-10.
- [33] N. Lasla, H. Ghazzai, H. Menouar and Y. Massoud, "Exploiting Land Transport to Improve the UAV's Performances for Longer Mission Coverage in Smart Cities," 2019 *IEEE 89th Vehicular Technology Conference (VTC2019-Spring)*, Kuala Lumpur, Malaysia, 2019, pp. 1-7.
- [34] Yu K, Budhiraja A K, Tokekar P. Algorithms for Routing of Unmanned Aerial Vehicles with Mobile Recharging Stations[J]. *Proceedings-IEEE International Conference on Robotics and Automation*, 2018: 5720-5725.
- [35] Ding N, Batta R, Kwon C. Conflict-free electric vehicle routing problem with capacitated charging stations and partial recharge[M]//*Technical Report*. Department of Industrial and Systems Engineering, University at Buffalo, US and Department of Industrial and Management Systems Engineering, University of South Florida, US, 2015.
- [36] Keskin M, Çatay B. Partial recharge strategies for the electric vehicle routing problem with time windows. *Transportation Research Part C*, 2016,65:111-127.
- [37] Erdelić, Tomislav; Carić, Tonči. A Survey on the Electric Vehicle Routing Problem: Variants and Solution Approaches[J]. *Journal of Advanced Transportation*, 2019.
- [38] Pisinger, D., Ropke, S., 2005. A General Heuristic for Vehicle Routing Problems. Technical Report, DIKU - Department of Computer Science, University of Copenhagen. http://www.diku.dk/hjemmesider/ansatte/sropke/Papers/GeneralVRP_TechRep.pdf (04.11.12).
- [39] Pisinger, D., Ropke, S., 2007. A general heuristic for vehicle routing problems[J]. *Computers & Operations Research* 34 (8), 2403–2435.
- [40] Ropke, S., Pisinger, D., 2006a. An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows[J]. *Transportation Science* 40(4), 455–472.
- [41] Demir E, Bekta T, Laporte G. An adaptive large neighborhood search heuristic for the Pollution-Routing Problem[J]. *European Journal of Operational Research*, 2012, 223(2):346-359.
- [42] Emeç Uğur, Çatay, Bülent, Bozkaya Burcin. An Adaptive Large Neighborhood Search for an E-grocery Delivery Routing Problem[J]. *Computers & Operations Research*, 2015: S0305054815002683.
- [43] Sacramento D, Pisinger D, Ropke S. An adaptive large neighborhood search metaheuristic for the vehicle routing problem with drones[J]. *Transportation Research*, 2019, 102(MAY):289-315.
- [44] Keskin M, Laporte G, Çatay B. Electric vehicle routing problem with time-dependent waiting times at recharging stations[J]. *Computers & Operations Research*, 2019, 107: 77-94.
- [45] Hiermann, Gerhard, Puchinger, Jakob, Ropke, Stefan, et al. The Electric Fleet Size and Mix Vehicle Routing Problem with Time Windows and Recharging Stations[J]. *European Journal of Operational Research*, 2016, 252 (3):995-1018.
- [46] Dorigo, M., Maniezzo, V., Colomi, A., Ant system: optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Mans, and Cybernetics*, 1996, vol. 1, no. 26, pp. 29–41.
- [47] Yu B, Yang Z Z, Yao B. An improved ant colony optimization for vehicle routing problem[J]. *European Journal of Operational Research*, 2009, vol. 196, no. 1, pp. 171-176.
- [48] Ding Q, Hu X, Sun L, et al. An improved ant colony optimization and its application to vehicle routing problem with time windows[J]. *Neurocomputing*, 2012, vol. 98, pp. 101-107.
- [49] Zhang S, Gajpal Y, Appadoo S S, et al. Electric vehicle routing problem with recharging stations for minimizing energy consumption[J]. *International journal of production economics*, 2018, 203:404-413.
- [50] D. C. Paraskevopoulos, P. P. Repoussis, C. D. Tarantilis, G. Ioannou, G. P. Prastacos. A reactive variable neighborhood tabu search for the heterogeneous fleet vehicle routing problem with time windows[J]. *Journal of Heuristics*, 2008, vol. 14, no. 5, pp. 425-455.
- [51] M. Bruglieri, F. Pezzell, O. Pisacane, and S. Suraci, "A variable neighborhood search branching for the electric vehicle routing problem with time windows[J]," *Electron. Notes Discrete Math.*, 2015, vol. 47, no. 15, pp. 221–228.
- [52] Huiting Mao, Jianmai Shi, Yuzhen Zhou, et al. The electric vehicle routing problem with time windows and multiple recharging options[J]. *IEEE Access*, 2020, PP (99):1-1.
- [53] Huiting Mao, Jianmai Shi. Experimental data set for UAV routing with recharging. https://www.researchgate.net/publication/346048168_Experimental_data_set_for_UAV_routing_with_recharging